CS6220: Data mining techniques
Multiple regression

Olga Vitek

September 24, 2015

Outline

Multiple regression

Example dataset

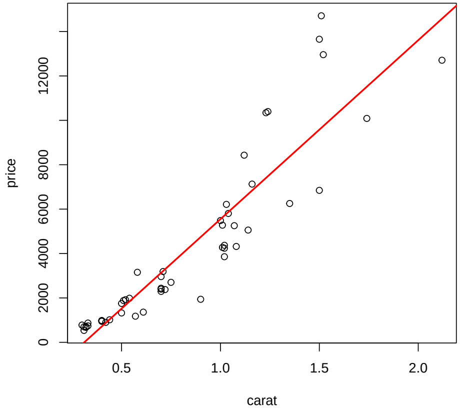
Example dataset:

- Let's take a random sample of 50 diamonds

```
> library(ggplot2)
> set.seed(123)
> index <- sample(1:nrow(diamonds), 50) # try a subset first
> diamonds2 <- diamonds[index,]
```

A simple linear regression

```
> plot(price ~ carat, data=diamonds2)
> abline(lm(price ~ carat, data=diamonds2), col='red', lwd=2)
```

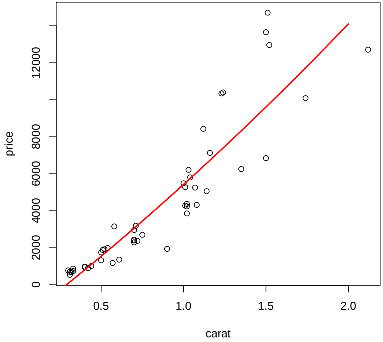


Summary of a simple linear regression

```
> summary(lm(price ~ carat, data=diamonds2))

Call:
lm(formula = price ~ carat, data = diamonds2)

Residuals:
    Min      1Q  Median      3Q     Max 
-2803.9  -913.7   -20.2   583.3  5049.3 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  -2511.5      502.6  -4.997 8.16e-06 ***
carat         8060.3      534.2  15.088  < 2e-16 ***
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ

Residual standard error: 1613 on 48 degrees of freedom
Multiple R-squared:  0.8259,	Adjusted R-squared:  0.8222 
F-statistic: 227.7 on 1 and 48 DF,  p-value: < 2.2e-16
```

Can a quadratic term help?

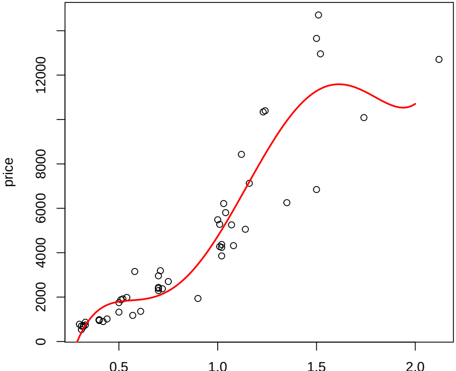
```
> plot(price ~ carat, data=diamonds2)
> newCarat=seq(from=0, to=2, length=100)
> lines(newCarat,
+       predict(lm(price ~ carat + I(carat^2), data=diamonds2),
+               newdata=data.frame(carat=newCarat)),
+       col='red', lwd=2)
```



The bias-variance trade-off

Can a very flexible polynomial help?

```
> plot(price ~ carat, data=diamonds2)
> newCarat=seq(from=0, to=2, length=100)
> lines(newCarat,
+     predict(lm(price ~ carat + poly(carat,5), data=diamonds2),
+               newdata=data.frame(carat=newCarat)),
+     col='red', lwd=2)
```

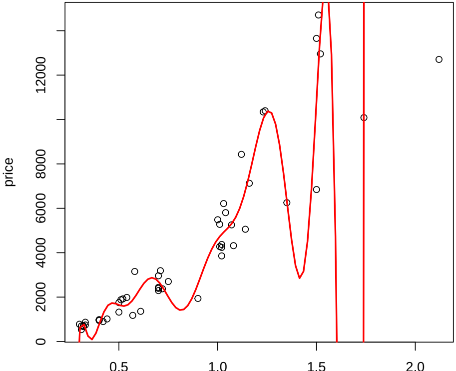


Can a very flexible polynomial help?

```
> summary(lm(price ~ carat + poly(carat, 5), data=diamonds2))

Call:
lm(formula = price ~ carat + poly(carat, 5), data = diamonds2)

Residuals:
    Min      1Q  Median      3Q     Max
-4438.7  -586.0   107.8   470.7  3372.2

Coefficients: (1 not defined because of singularities)
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      -2511.5      436.3  -5.756  7.7e-07 ***
carat             8060.3      463.7  17.382  < 2e-16 ***
poly(carat, 5)1       NA         NA      NA       NA
poly(carat, 5)2    973.3     1400.4   0.695 0.490686
poly(carat, 5)3  -5124.6     1400.4  -3.659 0.000673 ***
poly(carat, 5)4  -1315.5     1400.4  -0.939 0.352684
poly(carat, 5)5   3113.6     1400.4   2.223 0.031376 *
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.

Residual standard error: 1400 on 44 degrees of freedom
Multiple R-squared:  0.8797,     Adjusted R-squared:  0.8661
```

Can a very flexible polynomial help?

```
> plot(price ~ carat, data=diamonds2)
> newCarat=seq(from=0, to=2, length=100)
> lines(newCarat,
+       predict(lm(price ~ carat + poly(carat,15), data=diamonds2),
+               newdata=data.frame(carat=newCarat)),
+       col='red', lwd=2)
```

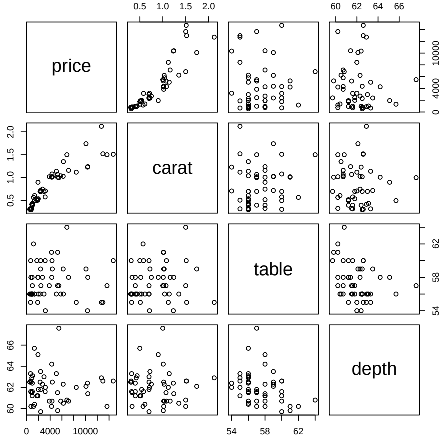


Conclusion

- More predictors can help build a better model (i.e., eliminate systematic bias)
- However, too many predictors overfit the data (i.e., introduce variance)
- Selecting the right number of predictors is the bias-variance trade-off

Select price + 3 quantitative descriptors

```
> library(dplyr)
> diamonds2 %>% select(price, carat, table, depth) %>% pairs()
```



Can 'table' explain additional variation in price?

```
> summary(lm(price ~ carat + table, data=diamonds2))

Call:
lm(formula = price ~ carat + table, data = diamonds2)

Residuals:
    Min      1Q  Median      3Q     Max
-2715.3  -940.4  -139.3   496.5  5425.7

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   7515.3     6061.9    1.24    0.221
carat         8165.7      528.5   15.45   <2e-16 ***
table         -176.0      106.1   -1.66    0.104
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.

Residual standard error: 1585 on 47 degrees of freedom
Multiple R-squared:  0.8355,        Adjusted R-squared:  0.8285
F-statistic: 119.4 on 2 and 47 DF,  p-value: < 2.2e-16
```

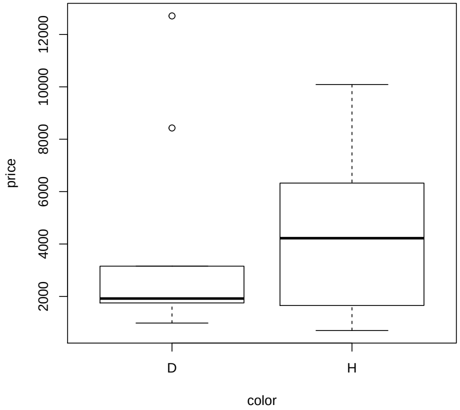
Can 'table' and 'depth' explain additional variation in price?

```
> summary(lm(price ~ carat + table + depth, data=diamonds2))

Call:
lm(formula = price ~ carat + table + depth, data = diamonds2)

Residuals:
    Min      1Q  Median      3Q     Max
-2642.2  -991.4  -130.7   466.3  5557.9

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  16246.4    13265.6   1.225   0.2269
carat         8168.7      531.1  15.381   <2e-16 ***
table         -201.4      112.0  -1.799   0.0786 .
depth         -117.3      158.4  -0.741   0.4625
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.

Residual standard error: 1592 on 46 degrees of freedom
Multiple R-squared:  0.8375,        Adjusted R-squared:  0.8269
F-statistic:    79 on 3 and 46 DF,  p-value: < 2.2e-16
```

Qualitative predictors

Qualitative predictors

```
> diamonds3 <- subset(diamonds2, color=='D' | color == 'H')
> diamonds3$color <- factor(diamonds3$color, ordered=FALSE)
> plot(price ~ color, data=diamonds3)
```



Qualitative predictors

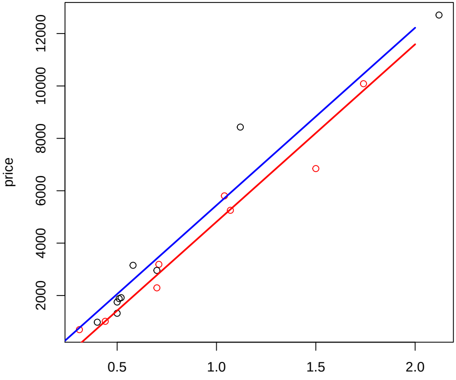
```
> summary(lm(price ~ color, data=diamonds3))

Call:
lm(formula = price ~ color, data = diamonds3)

Residuals:
   Min     1Q Median     3Q    Max
 -3701  -2149  -1209   1406   8806

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3901.0     1215.2   3.210  0.00584 **
colorH         497.6     1771.5   0.281  0.78262
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.

Residual standard error: 3646 on 15 degrees of freedom
Multiple R-squared:  0.005233,       Adjusted R-squared:  -0.06108
F-statistic: 0.07891 on 1 and 15 DF,  p-value: 0.7826
```

Qualitative predictors

```
> summary(lm(price ~ carat + color, data=diamonds3))

Call:
lm(formula = price ~ carat + color, data = diamonds3)

Residuals:
    Min      1Q  Median      3Q     Max
-1356.5  -396.3  -241.8   341.7  2171.9

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -1332.7      412.9  -3.227  0.00608 **
carat         6777.4      401.8  16.866 1.07e-10 ***
colorH        -631.0      402.7  -1.567  0.13947
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.

Residual standard error: 817.3 on 14 degrees of freedom
Multiple R-squared:  0.9533,        Adjusted R-squared:  0.9467
F-statistic:   143 on 2 and 14 DF,  p-value: 4.815e-10
```

Qualitative predictors imply parallel lines for each color

```
> plot(price ~ carat, col=diamonds3$color, data=diamonds3)
> newCarat=seq(from=0, to=2, length=100)
> lines(newCarat, predict(lm(price ~ carat + color, data=diamonds3),
+  newdata=data.frame(carat=newCarat, color=rep('H', 100))), col='red',
> lines(newCarat, predict(lm(price ~ carat + color, data=diamonds3),
+  newdata=data.frame(carat=newCarat, color=rep('D', 100))), col='blue'
```



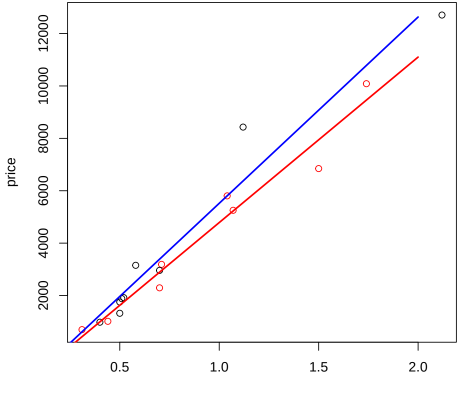
Statistical interaction

Qualitative predictors

```
> summary(lm(price ~ carat*color, data=diamonds3))

Call:
lm(formula = price ~ carat * color, data = diamonds3)

Residuals:
    Min      1Q  Median      3Q     Max
-1098.0  -427.4  -188.4   271.1  2055.9

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -1590.36     491.08  -3.238  0.00647 **
carat         7111.11     528.67  13.451 5.26e-09 ***
colorH          58.99     815.43   0.072  0.94343
carat:colorH  -794.21     815.64  -0.974  0.34797
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.

Residual standard error: 818.8 on 13 degrees of freedom
Multiple R-squared:  0.9565,	Adjusted R-squared:  0.9465
F-statistic: 95.31 on 3 and 13 DF,  p-value: 4.206e-09
```

Statistical interaction allows non-parallel lines

```
> plot(price ~ carat, col=diamonds3$color, data=diamonds3)
> newCarat=seq(from=0, to=2, length=100)
> lines(newCarat, predict(lm(price ~ carat*color, data=diamonds3),
+  newdata=data.frame(carat=newCarat, color=rep('H', 100))), col='red',
> lines(newCarat, predict(lm(price ~ carat*color, data=diamonds3),
+  newdata=data.frame(carat=newCarat, color=rep('D', 100))), col='blue'
```



Multicollinearity

Including correlated predictors is not helpful

```
> summary(lm(price ~ x, data=diamonds2))

Call:
lm(formula = price ~ x, data = diamonds2)

Residuals:
    Min      1Q  Median      3Q     Max
-2916.8 -1297.6  -120.8   874.9  6015.5

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   -14989       1524  -9.837 4.32e-13 ***
x               3280        256  12.812  < 2e-16 ***
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.

Residual standard error: 1839 on 48 degrees of freedom
Multiple R-squared:  0.7737,       Adjusted R-squared:  0.769
F-statistic: 164.2 on 1 and 48 DF,  p-value: < 2.2e-16
```

Including correlated predictors is not helpful

```
> summary(lm(price ~ y, data=diamonds2))

Call:
lm(formula = price ~ y, data = diamonds2)

Residuals:
    Min      1Q  Median      3Q     Max
-2737.3 -1396.8   -78.0   990.5  5811.7

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -14923.6     1512.2  -9.869 3.89e-13 ***
y             3272.1      254.3  12.867  < 2e-16 ***
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.

Residual standard error: 1833 on 48 degrees of freedom
Multiple R-squared:  0.7753,	Adjusted R-squared:  0.7706
F-statistic: 165.6 on 1 and 48 DF,  p-value: < 2.2e-16
```

Including correlated predictors is not helpful

```
> summary(lm(price ~ x+y, data=diamonds2))

Call:
lm(formula = price ~ x + y, data = diamonds2)

Residuals:
    Min      1Q  Median      3Q     Max
-2755.3 -1380.0   -71.8   977.9  5831.1

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   -14934       1538  -9.712 8.15e-13 ***
x                328       5240   0.063    0.950
y               2946       5222   0.564    0.575
---
Signif. codes:  0 âĂŸ***âĂŹ 0.001 âĂŸ**âĂŹ 0.01 âĂŸ*âĂŹ 0.05 âĂŸ.âĂŹ 0.

Residual standard error: 1852 on 47 degrees of freedom
Multiple R-squared:  0.7753,      Adjusted R-squared:  0.7657
F-statistic: 81.07 on 2 and 47 DF,  p-value: 5.808e-16
```

Including correlated predictors is not helpful

```
> diamonds2 %>% select(x,y,z) %>% cor

          x         y         z
x 1.0000000 0.9987886 0.9895721
y 0.9987886 1.0000000 0.9893948
z 0.9895721 0.9893948 1.0000000

> library(car)
> vif(lm(price ~ x+y+z, data=diamonds2))

       x        y        z
428.9020 421.7663  49.2219
```